

# Particle Swarm Optimization: Application to Nanostructure Reconstruction from X-Ray Scattering Data

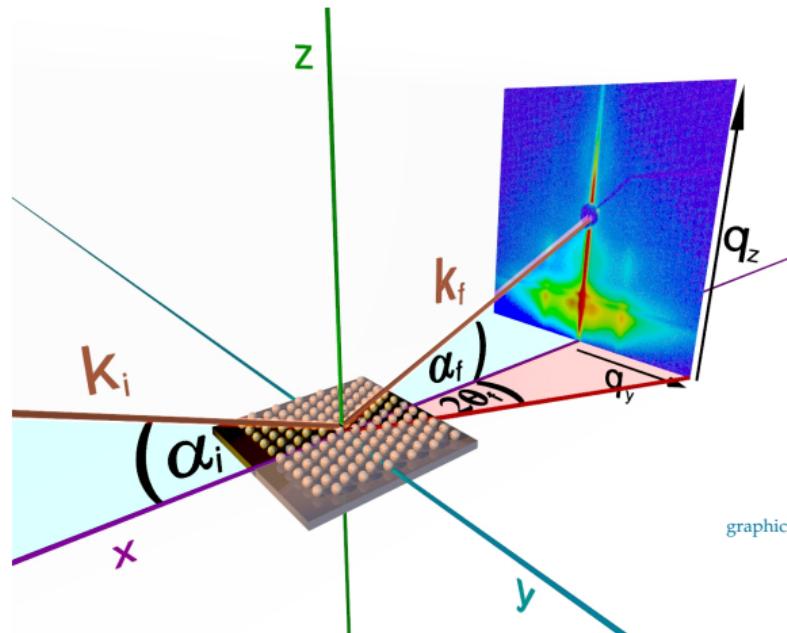
Abhinav Sarje

Computational Research Division  
Lawrence Berkeley National Laboratory



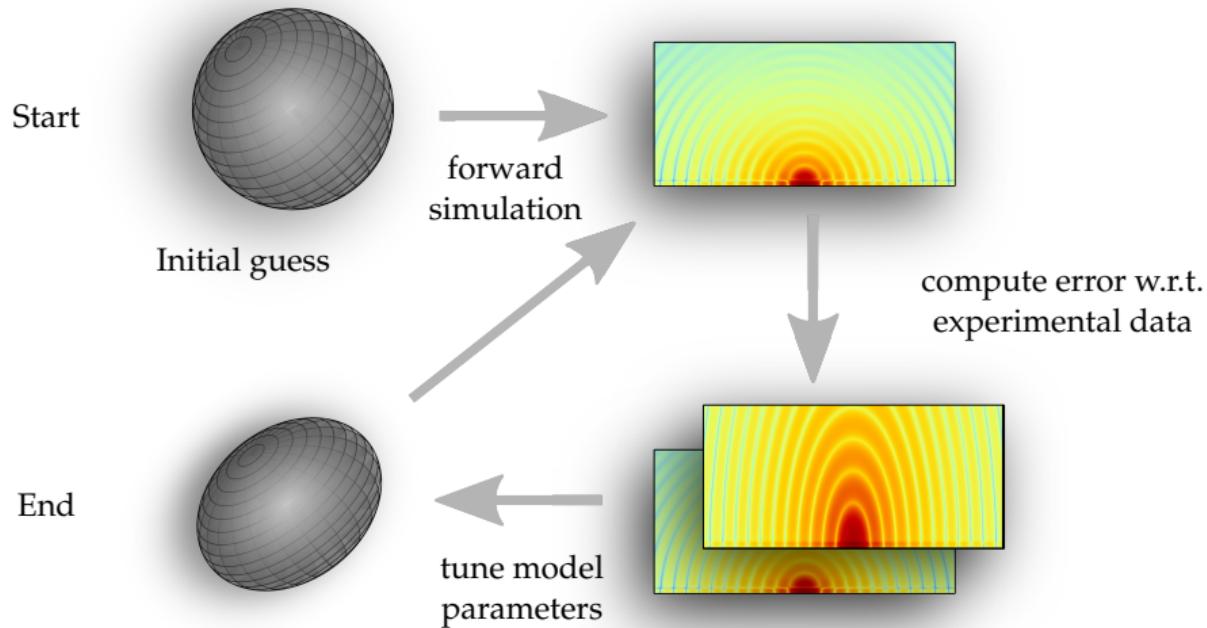
December 09, 2015

# Small Angle X-ray Scattering at Synchrotrons



graphic: courtesy of A. Meyer, [www.gisaxs.de](http://www.gisaxs.de)

# Inverse Modeling for Reconstruction: The Optimization Problem



# Inverse Modeling for Reconstruction

Various optimization algorithms exist. Some of the representative algorithms in our case:

- Stochastic: Reverse Monte-Carlo.
- Gradient based: LMVM (Limited-Memory Variable-Metric.)
- Derivative-free trust region-based: Pounders.
- Stochastic: Particle Swarm Optimization.

Objective functions: Forward simulation for computing scattered light intensities. E.g.

- Fast Fourier Transformation computations.
- Complex form factor and structure factor computations.

## The Objective Function: Forward Simulations

Given:

- ① a sample structure model, and
- ② experimental configuration,

simulate scattering patterns.

Scattered light intensity is computed at each point  $\vec{q}$  in the detector inverse space.

## The Objective Function: Forward Simulations

- Intensity is proportional to square of the sum of *Form Factors* (and *Structure Factors*) at  $\vec{q}$ :

$$I(\vec{q}) \propto \left| \sum_{s=1}^S F(\vec{q}) \right|^2$$

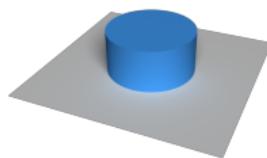
- Form Factor at  $\vec{q}$  is an integral over an object surface.

$$F(\vec{q}) = -\frac{i}{|\vec{q}|^2} \int_{S(\vec{r})} e^{i\vec{q}\cdot\vec{r}} q_n(\vec{r}) d^2\vec{r}$$

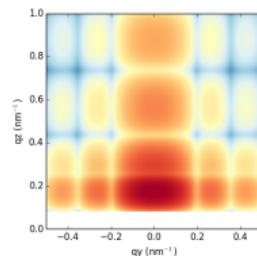
- Computed as summation over a discretization of the object surface:

$$F(\vec{q}) \approx -\frac{i}{|\vec{q}|^2} \sum_{k=1}^t e^{i\vec{q}\cdot\vec{r}_k} q_{n,k} \sigma_k$$

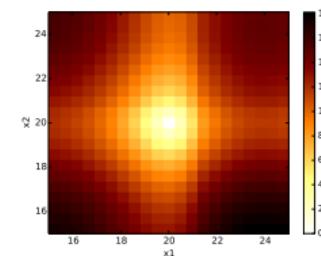
# LMVM and POUNDerS: Two Parameter Case



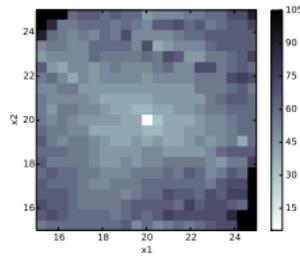
A Single Cylindrical Nanoparticle



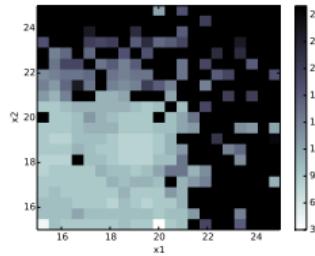
X-Ray Scattering Pattern



Objective Function Map

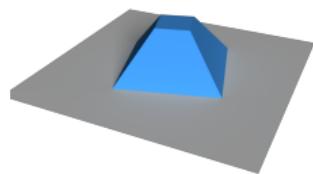


LMVM Convergence Map

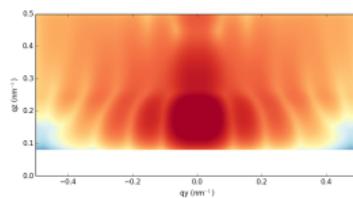


POUNDerS Convergence Map

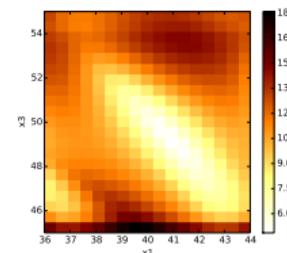
# LMVM and POUNDerS: Six Parameter Case



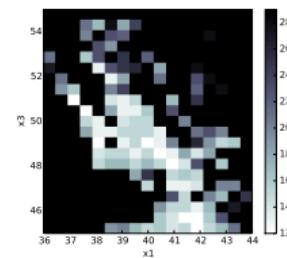
Pyramidal Nanoparticles  
forming a Lattice



X-Ray Scattering Pattern



Objective Function Map



POUNDerS Convergence Map

LMVM does not converge

# Particle Swarm Optimization

- A class of stochastic methods.
- Multiple agents, “*particle swarm*”, evaluate for optimal positions within the parameter search space.
- Agents perform quasi “random-walks”.
- Agent “velocities” (displacements) are influenced by the histories of the traveled paths.
- No initial guesses of parameter values necessary.

# Particle Swarm Optimization

$$\vec{v}_i \leftarrow \omega \vec{v}_i + (\vec{b}_i - \vec{x}_i) r_1 \phi_1 + (\vec{b}_g - \vec{x}_i) r_2 \phi_2$$

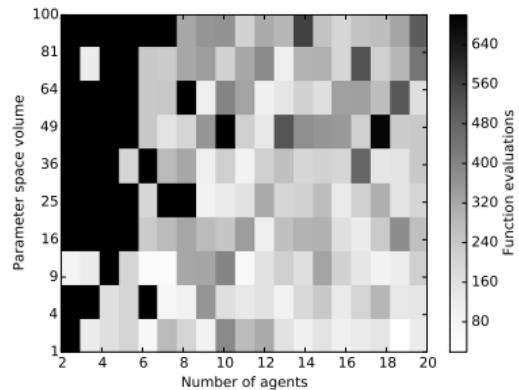
inertia coefficient

force coefficients

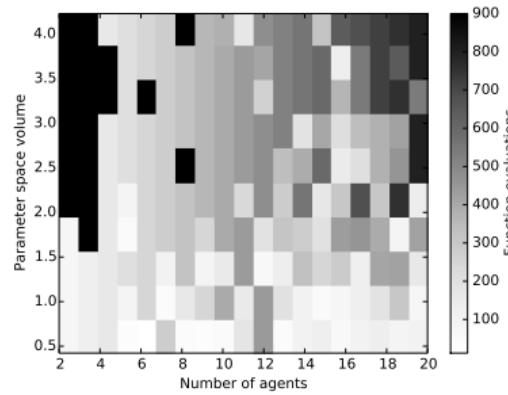
local best position

global best position

# Particle Swarm Optimization: Fitting X-Ray Scattering Data

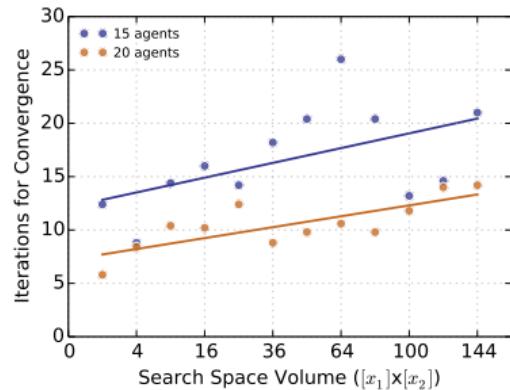


Fitting 2 Parameters

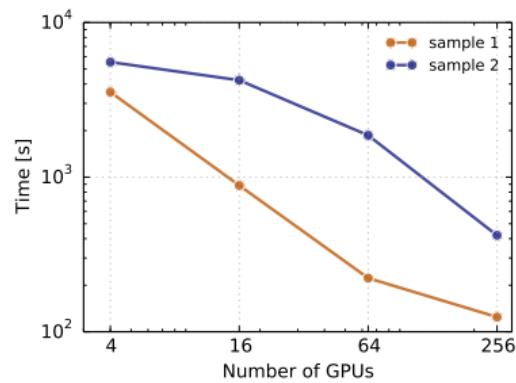


Fitting 6 Parameters

# Particle Swarm Optimization: Runtime Performance

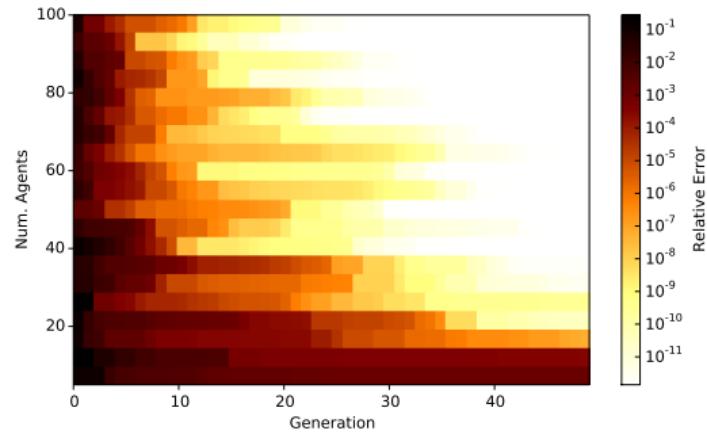
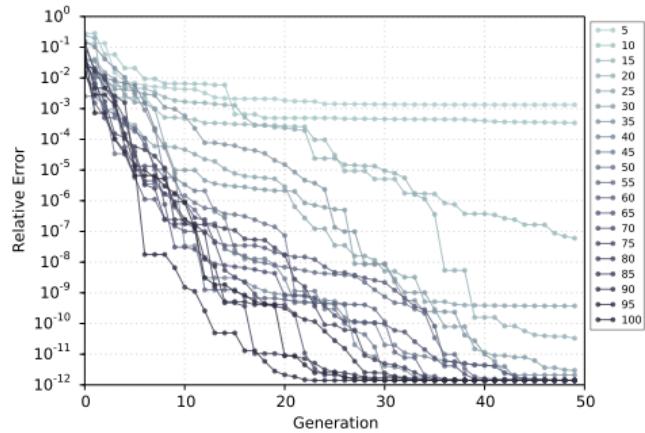


Convergence w.r.t. Search Space Volume



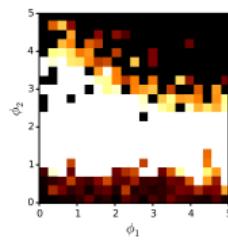
Strong Scaling on Titan

# Particle Swarm Optimization: Agents vs. Generations

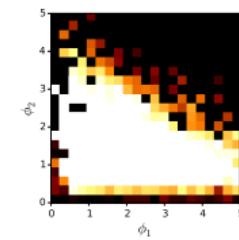


# Deeper into Particle Swarm Optimization

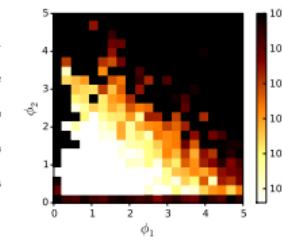
Selecting coefficient values  $\omega, \phi_1, \phi_2$ :



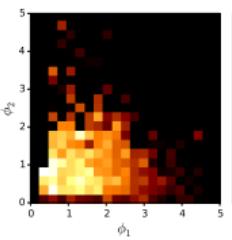
$$\omega = 0.00$$



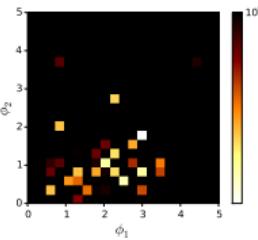
$$\omega = 0.25$$



$$\omega = 0.50$$



$$\omega = 0.75$$



$$\omega = 1.00$$

## Particle Swarm Optimization: Variants

Numerous variations of the basic PSO algorithm exist. Each is a variant in either:

- Agent's position displacement computation method.
- Connection topology among the agents.
- Combination of both.

$$\vec{v}_i \leftarrow \omega \vec{v}_i + (\vec{b}_i - \vec{x}_i) r_1 \phi_1 + (\vec{b}_g - \vec{x}_i) r_2 \phi_2$$

Diagram illustrating the components of the Particle Swarm Optimization velocity update equation:

- inertia coefficient:  $\omega$
- local best position:  $\vec{b}_i$
- global best position:  $\vec{b}_g$
- force coefficients:  $r_1 \phi_1$  and  $r_2 \phi_2$

## Particle Swarm Optimization: Basic

- ① Each agent evaluates the objective function at its current position in the search space.
- ② Each agent identifies its “local” best position yet.
- ③ With fully-connected topology, all agents collaborate to find the “global” best position yet.
- ④ Each agent computes its displacement and moves to the new position.

# Particle Swarm Optimization: Position Displacement

Inertia coefficient  $\omega$ , as

- a constant value throughout,
- damped value through constriction coefficient  $\chi$ ,
- a variable and/or tuned value.

Fully-informed particle swarm (FIPS).

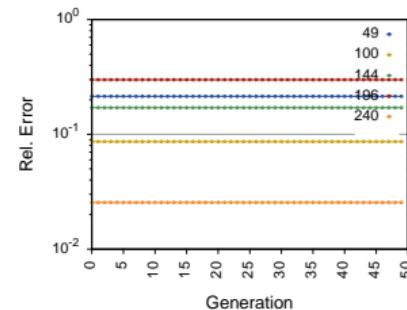
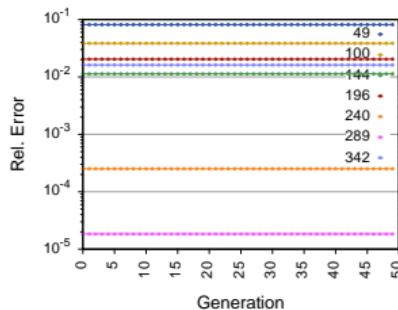
- Each agent explicitly influences all other agents (all-to-all).

Fitness distance ratio (FDR).

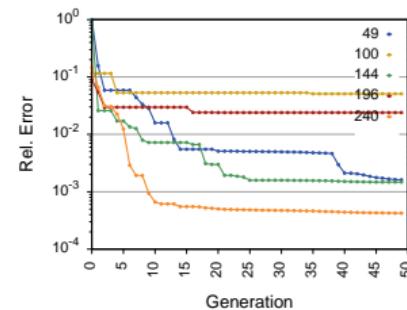
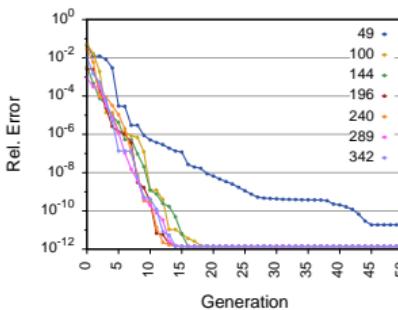
- Replace Euclidean distance measure between agents' position from local and global best with a distance ratio using the corresponding "fitness" values.

# Particle Swarm Optimization: Position Displacement

FIPS:



FDR:



## Particle Swarm Optimization: Topology

Position updates are influenced only by the “neighborhoods” defined by the topology used.

Lbest.

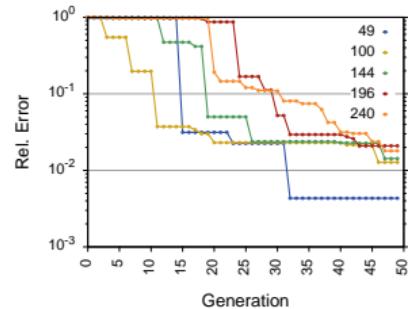
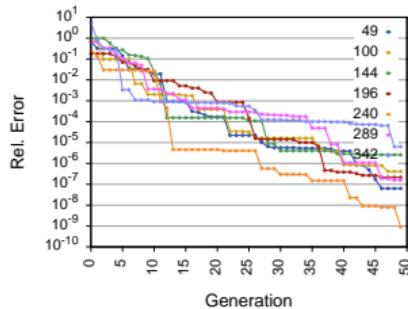
- Ring lattice topology – each agent has exactly two neighbors.
- Increases agent independence making them less susceptible by others’ local minima.
- Takes longer to converge, but in general provides better solutions.
- Also, avoids global communication.

von Neumann topology.

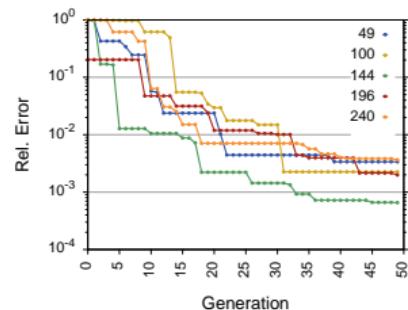
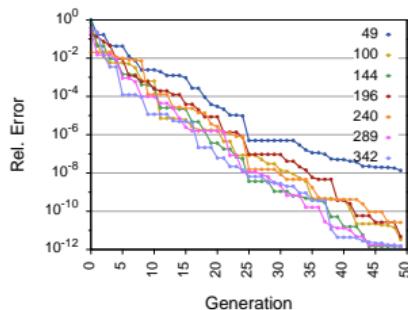
- Torus lattice topology – each agent has exactly four (2D) or six (3D) neighbors.
- Better balance between Lbest and fully-connected topologies.

# Particle Swarm Optimization: Topology

Lbest:

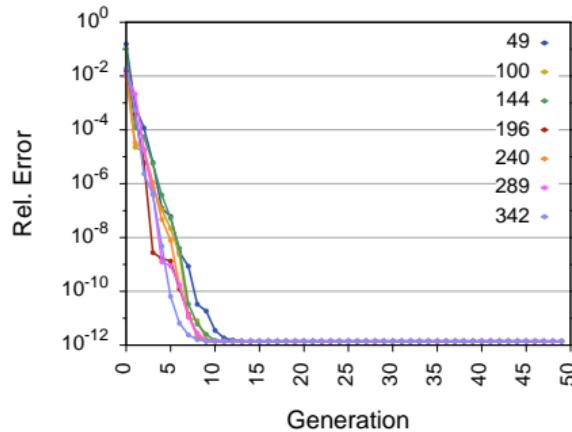


von Neumann:



## Particle Swarm Optimization: Look-Ahead

- Better exploit massive parallelism.
- Each agent identifies best of  $k$  possible positions from its current position.
- Uses more comprehensive knowledge of search space neighborhood to identify its path.



# An Ongoing Work ...

We saw that

- Gradient based methods converge only for simple cases, and gradient-free trust region based methods are highly sensitive to the initial guess.
- PSO works without the need of an initial guess.
- PSO is robust, nearly always converging, but computationally expensive.

Near future:

- Evaluating other stochastic methods.
- Using machine learning for feature and structural classification to generate initial models to fit.
- Evaluating neuromorphic computing for classification.

# An Ongoing Work ...

We saw that

- Gradient based methods converge only for simple cases, and gradient-free trust region based methods are highly sensitive to the initial guess.
- PSO works without the need of an initial guess.
- PSO is robust, nearly always converging, but computationally expensive.

Near future:

- Evaluating other stochastic methods.
- Using machine learning for feature and structural classification to generate initial models to fit.
- Evaluating neuromorphic computing for classification.

## Further Information

- ① **HipGISAXS: a high-performance computing code for simulating grazing-incidence X-ray scattering data.** Journal of Applied Crystallography, vol. 46, pp. 1781–1795, 2013.
- ② **Massively parallel X-ray scattering simulations.** Supercomputing (SC), 2012.
- ③ **Tuning HipGISAXS on multi- and many-core supercomputers.** Performance Modeling, Benchmarking and Simulation of High Performance Computer Systems, 2013.
- ④ **High performance inverse modeling with Reverse Monte Carlo simulations.** International Conference on Parallel Processing, 2014.

## Acknowledgments

- Thanks to NVIDIA for donating several GPU cards to make this work possible.
- Supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.
- Also supported by DoE Early Career Research grant awarded to Alexander Hexemer.
- Used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.
- Used resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

Thank you!

Contact: [asarje@lbl.gov](mailto:asarje@lbl.gov)